Effect of Time Derivatives of MFCC Features on HMM Based Speech Recognition System

Sanghamitra V. Arora[1]

[1] K.J. Somaiya College of Engineering, Electronics and Telecommunication Department, Mumbai, India

Email: arora.sanghamitra@gmail.com

Abstract— **In this paper, improvement of an ASR system for Hindi language, based on Vector quantized MFCC as feature vectors and HMM as classifier, is discussed. MFCC features are usually pre-processed before being used for recognition. One of these pre-processing is to create delta and delta-delta coefficients and append them to MFCC to create feature vector. This paper focuses on all digits in Hindi (Zero to Nine), which is based on isolated word structure. Performance of the system is evaluated by accurate Recognition Rate (RR). The effect of the combination of the Delta MFCC (DMFCC) feature along with the Delta-Delta MFCC (DDMFCC) feature shows approximately 2.5% further improvement in the RR, with no additional computational costs involved. RR of the system for the speakers involved in the training phase is found to give better recognition accuracy than that for the speakers who were not involved in the training phase. Word wise RR is observed to be good in some digits with distinct phones.**

Index Terms— **Automatic Speech Recognition (ASR), Mel Frequency Cepstral Coefficient (MFCC), Delta MFCC (DMFCC), Delta Delta MFCC (DDMFCC), Hidden Markov Model (HMM), Recognition Rate (RR).**

I. Introduction

For several decades, designing an interactive machine that can imitate human behavior, specially the capability of understanding and making complex decisions and that too in a speed that matches human brain has attracted a great deal of interest. This can be achieved by using the power of computers to develop Automatic Speech Recognition (ASR) system, which captures a relevant input signal and transforms it into written text. Textual translation of speech signal is the most common objective of ASR, but these systems can also support spoken queries, dictation systems, command and control medical applications and speech translation from one language to the other. Most of the systems developed till date are based on English and other foreign language speech, which restricts the usage of these machine oriented speech based interface only among educated Hindi speaking population of India. It is a fact that effective communication always takes place in speaker's own language; hence speech interface for database queries supporting Hindi is a special need. Very limited work has been done in support of Hindi and other Indian languages.

Since each individual has a different vocal tract controlled by his brain, speakers following languages based on same linguistic rules have different speaking style, rate and accent. Even same word repeated by same speaker has variations, which can be readily observed in its digital representation. Technological advancements have tried to deal with such type of variability and also to reduce the complexity of speech systems. With major technological advancements in dealing with temporal and spectral variability in speech signals, ASRs with capability to recognize speech independent of speakers have been developed.

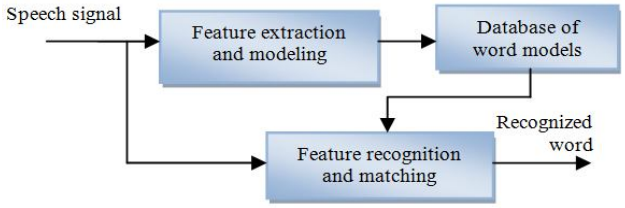

Speech signal
Feature extraction and modeling
Database of word models
Recognized word
Feature recognition and matching

Figure 1. Speech recognition system

Vocal tract information like formant frequency, bandwidth of formant frequency and other values may be linked to an individual person. The goal of a feature extraction block technique is to characterize the information, as shown in Figure. 1. A wide range of possibilities exists for parametrically representing the speech signal to be used for Speaker independent speech recognition purpose. Some of the common techniques used are: Linear Prediction Coding (LPC); Mel-Frequency Cepstral Coefficients (MFCC); Linear Predictive Cepstral Coefficients (LPCC); Perceptual Linear Prediction (PLP); and Neural Predictive Coding (NPC) [1], [2] and [4]. MFCC is a popular technique because it is based on the known variation of the human ear's critical frequency bandwidth. MFCC coefficients are obtained by de-correlating the output log energies of a filter bank which consists of triangular filters, linearly spaced on the Mel frequency scale. Conventionally an implementation of discrete cosine transform (DCT) known as is used to de-correlate the speech. The acoustic vectors can be used as feature vectors. It is possible to obtain more detailed speech features by using a derivative on the MFCC acoustic vectors [10] and [11]. This temporal approach permits the computation of the delta MFCC (DMFCCs), as the first order derivatives of the MFCC. Then, the delta-delta MFCC (DDMFCCs) are derived from DMFCC, being the second order derivatives of MFCCs. Feature selection is followed by Vector quantization (VQ) on the derived MFCC coefficients, using LBG algorithm [6]. VQ algorithm saves a lot of time during the testing phase as we were only considering few feature vectors instead of overloaded feature space of a particular user**.** The word model was generated using Hidden Markov Model (HMM), which is a state machine [1] and [2].

This paper initiates with the discussion on the overall framework of the HMM based ASR system, in the next section it covers signal processing techniques extending it from feature extraction methods to statistical approach used for speech modeling. Challenges in the ASR depending on the phonetic characteristics of the spoken words, conclusion and future scope have been covered in subsequent sections.

II. Signal Processing for Asr

A. *Pre-processing*

The overall ASR is executed in two phases: Pre-processing and post processing phase. During pre-processing, noise and unwanted speech segment is removed and further features are extracted from the clean speech. In post processing phase, speech recognition engine is developed which is based upon the knowledge of acoustic and language model. The model so obtained is used by the recognition engine to correctly identify the most likely match for the test input. For a small vocabulary system, building a model for a word required us to collect the sound files of the word from various users. These sound files were then used to train a HMM Model. The isolated word recognition system may be viewed as working in the following stages:

1. Sound Recorder: The component was responsible for taking input from microphone and its output is a wave file or a direct feed for the feature extractor.

2. Pre emphasis: A pre emphasis filter was used to eliminate the -6dB per octave decay of the spectral energy and also to boost the high frequency signals. Word detection was done using energy and zero crossing rate of the signal.

3. Feature Extraction: The usual objective in selecting a parametric representation of speech is to compress it by eliminating information that are not crucial for phonetic analysis of speech data while enhancing those aspects of the signal that contribute most to the detection of phonetic differences [2] and [8]. Psychophysical studies have shown that human perception of the frequency contents of sounds for speech signals does not follow a linear scale. The cepstral representation of the speech spectrum on a scale called the 'mel' scale provided a good representation of the local spectral properties of the signal for the given frame analysis. Filters are spaced linearly at low frequencies and logarithmically at high frequencies to capture the important characteristics of speech. The Mel-frequency scale is linearly spaced at frequency below 1000 Hz and logarithmically spaced at frequency above 1000 Hz. Figure. 2 represents the block diagram of MFCC feature extraction. Speech signal is divided into overlapping frames of 20-25 ms and windowed, typically using Hamming window to remove discontinuity at the frame boundary. FFT (Conversion from time domain to frequency domain) is obtained for each frame and is passed through set of triangular filters spaced according to the perceptual Mel scale (Mel-Frequency warping). Logarithm of spectral amplitude is then taken and the output so obtained is finally converted back to time domain by an inverse FFT (Cepstrum) process to give MFCCs.

Because the mel spectrum coefficients are real numbers, we converted them to the time domain using the Discrete - Cosine Transform (DCT). The final set of MFCC coefficients is called an acoustic vector [1,2,4 and 8]. Mel Frequency Cepstral Coefficients (MFCC) were extracted for each word and statistical conditioning of these vectors were done to form observation vectors as shown in Figure. 2. Therefore each input utterance is transformed into a sequence of acoustic vectors.

$$\mathrm{c_n} = \sum_{k=1}^{k} \log \mathrm{G}\,(k) \cos[n(k-1/2)\pi/k] \qquad n = 1,2 \ldots k^{\square}$$

The features outlined above don't have temporal information. In order to incorporate the ongoing changes over multiple frames, time derivatives are added to the basic feature vector. The first and second derivatives of the feature are usually called Delta coefficients and Delta-Delta coefficients respectively. The Delta coefficients are computed via a linear regression formula:

$$\Delta c[m] = \frac{\sum_{m}^{k} i(c\,[m+i] - c[m-i])}{2\sum_{i=1}^{k} i^2}$$

where, (2k+1) is the size of the regression window and c[m] is the m th MFCC coefficient. The Delta-Delta coefficients are computed using linear regression of Delta features.

A typical speech recognition system has a 39-element feature vector. The feature vector consists of 13 static features (12 MFCCs computed from 24 filter banks and energy), 13 delta coefficients (first derivatives of static features) and 13 delta-delta coefficients (second derivatives of static features). The complete feature extraction procedure for a typical speech recognition system is as shown in Figure 2.

We note that the addition of delta-cepstral features to the static 13-dimensional MFCC features strongly improves speech recognition accuracy, and a further (smaller) improvement is provided by the addition of double-delta

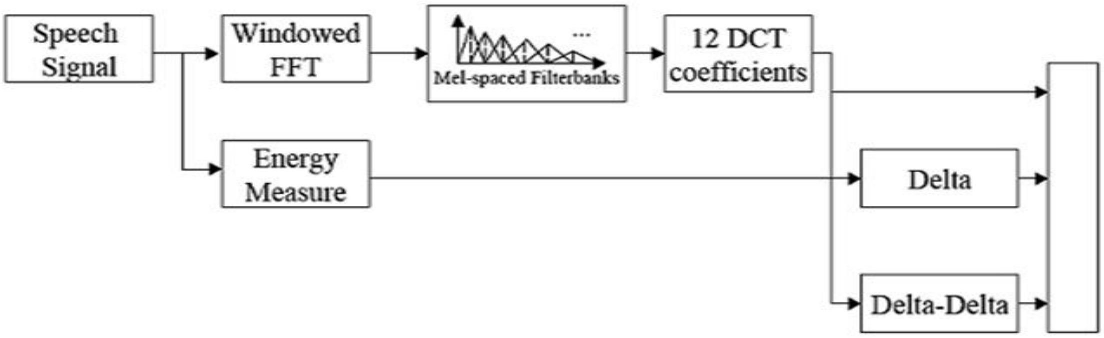


Figure 2. Feature extraction with temporal details.

© 2013 ACEEE
DOI: 01.IJSIP.4.3.1257

cepstral features. For these reasons some form of delta and double-delta cepstral features are part of nearly all speech recognition systems, so the delta-features are uncorrelated with the static features and help the frame independence assumption in the HMM in ASR. The addition of delta-cepstral coefficients (DCC) to MFCC coefficients improves ASR recognition accuracy [10] and [11].

B. Post-processing

1. Vector quantization: It is a process of mapping vectors from a large vector space to a finite number of regions in that space. Each region is called a cluster and can be represented by its center called a centroid. Vector quantization on the MFCC coefficients was done using LBG algorithm as shown in Figure. 3 [3,6 and 8].

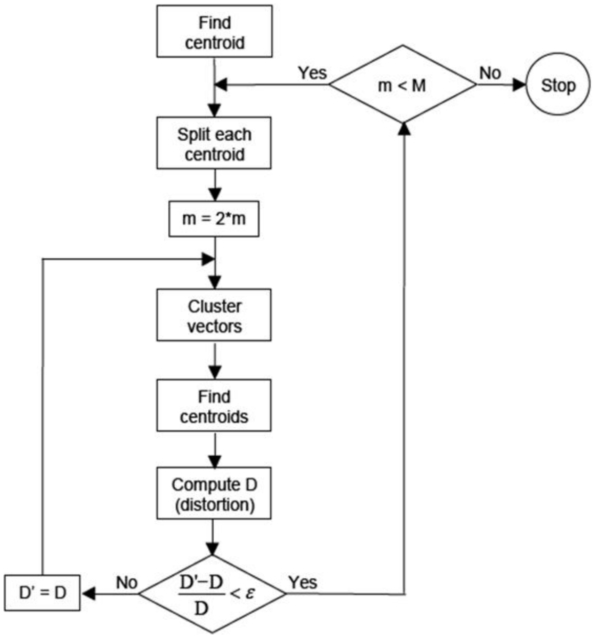


Figure 3. The flow chart of the LBG algorithm

VQ algorithm saves a lot of time during the testing phase as we were only considering few feature vectors instead of overloaded feature space of a particular user [8]. The size of the codebook is increased by splitting each current codebook $\mathbf{y}_n$ according to the rule,

$$\mathbf{y}_n^+ = \mathbf{y}_n(1+\varepsilon)$$
$$\mathbf{y}_n^- = \mathbf{y}_n(1-\varepsilon)$$

where *n* varies from 1 to the current size of the codebook and 'ε' is a splitting parameter (we choose? ε = 0.01). Therefore is an economical compromise for discrete representation of speech sounds.

2. *Model Generation:* The word model was generated using Hidden Markov Model (HMM), which is a state machine [1]. HMM Model is defined as: (Q,O,A,B, ð)

i) $Q_i = \{q_i\}$ (all possible states)

ii) $O = \{v_i\}$ (all possible observations),

iii) $A = \{a_{ij}\}$, where $a_{ij} = P(X_{t+1} = q_j \mid X_t = q_i)$ (transition probabilities),

iv) $B = \{b_i\}$, where $b_i(k) = P(O_t = v_k \mid X_t = q_{it})$ (observation probabilities of observation k at state i),

v) {ð$_i$} = $P(X_0 = q_i)$ (initial state probabilities),

X_t denotes the state at time t and O_t denotes the observation at time. A version of the EM algorithm called the Baum-Welch algorithm is used to solve the learning problem.

III. Analytical Background For Hidden Markov Model

HMM is a network representation of acoustic events based on their statistical information in speech. First order Markov chain is considered as the base for this model, where the likelihood of being in a given state depends only on the immediate prior state. These networks are called HMM because the models are inferred through observation of speech output, not from any internal representation of speech production [3].

HMM works as a finite state machine which is assumed to be built up from a finite set of possible states having some probability density function (PDF) associated with them. Fundamental problems of HMMs are probability evaluation, determination of the best sequence, and parameter estimation. Several search algorithms and methods are available for probability evaluation and for finding the best sequence. For parameter estimation in ASR HMM uses maximum likelihood estimation (MLE) using a forward-backward procedure [5] and [7]. At present, much of the recent researches on speech recognition involve recognizing isolated or continuous speech from a large vocabulary using HMMs or a hybrid of HMMs.

A process in the HMM class can be described as a finite-state Markov Chain with a memory less output process which produces symbols in a finite alphabet. HMM-s are used for modeling of word(s) pronounced by one or more speakers, where modeling means obtaining a HMM to give a maximum resemblance probability only for the input word it was trained for. The model are first order left to right Markov model with N states (N=3), Bakis model [3] in our implementation. When a sequence $O = (O_1, O_2, \ldots.. O_t)$, in our case consisting of quantized cepstral vectors, feed the HMM input, the conditional probability $P(\boldsymbol{O}\,\boldsymbol{\lambda})$ is expected at the HMM output.

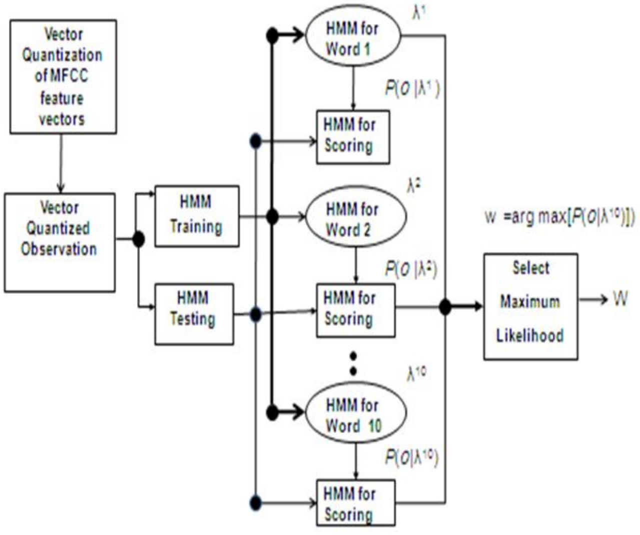


Figure 4. Block diagram of HMM Training and Testing Model

A. Training of HMM

Baum-Welch algorithm for learning:

In a HMM based recognition system, a separate HMM is built (trained) for each word *O*, *O* ε *W* , *W* the set under recognition as shown in Figure. 4. The training of each HMM consists in setting up of its matrices *A* and *B*, so that the probability $P(\boldsymbol{O} \mid \lambda)$ to be maximum. Baum proposed a monotone converging algorithm for recursive adjustment of HMM. Here, instead of summation over all possible paths, $P(\boldsymbol{O} \mid \lambda)$ was evaluated only over the maximal path. Most challenging of all was to adjust the model parameter (A,B, λ) to maximize the probability of the observation sequence given the model. Given any finite observation sequence as training data, there is no optimal way of estimating the model parameters.

Re-estimation Procedure:

In order to define a procedure for re-estimation of HMM parameter, we first define

$$\xi_t\ (i,j) = P(q_t = S_i\ , q_{t+1} = Sj\,|\lambda)$$

i.e., the probability of being in state S_i at time t, and state S_j at time t+1 given the model and observation sequence we can write

$$\xi_t\ (i,j) = \frac{\alpha_t(i)\alpha_{ij}\, b_j(O_{t+1})\ \beta_{t+1}\ (j)}{P(O|\lambda)}$$

Where $\beta_t\ (j$ = backward variable, i.e., probability of the partial observation sequence t + 1 to end (T), given state S_i at time t and model. Now we defir $\gamma_t(i)$ as the probability of being in state S_i at time t, given the observation sequence and model. Hence we can relate $\gamma_t(i\ )$ to ξ_t (i,j) by summing over j, such that,

$$\gamma_t(i) = \sum_{j=1}^{N} \xi_t(i,j)$$

If we sum $\gamma_t(i)$ over time index we get a quantity which can be interpreted as expected number of time state S_i is visited, or expected number of transition made from S_i. Similarly summation of ξ_t (i,j) over time can be interpreted as the expected number of transition made from state S_i to state S_j.The required three parameters of the HMM can be obtained by iterative procedure**.**

B. Transition matrix modeling

Often for the $P(\boldsymbol{O}|\lambda)$ calculation, the output probabilities *B* matrix elements for given state S_i of HMM, are modeled by Q-dimensional Gaussian mixture density G(μ_i ,”$_i$), where, μ_i is the mean of all cepstral vectors O_t that are generated in the state $S_{i.}$ T he covariance matrix, “$_1$ is most often assumed diagonal one, i.e. that all cepstral vector components $O_t\, \varepsilon\, O$ are uncorrelated. Thus each model G (μ_i ,”$_i$) is independently examined and can be modeled by the multiplication:

$$b_i(o_t) = G(o_t\,;\mu_i,\sigma_i) = \prod_{q=1}^{Q} \square\, G(o_t\,;\mu_i,\sigma_i)$$

$$= \frac{1}{\sqrt{(2\pi)^Q}} \prod_{q=1}^{Q} \frac{1}{\sigma_{iq}} \exp\left(-\frac{(O_t-\mu_{iq})^2}{2\sigma_{iq}^2}\right)$$

As a HMM is trained for given word, its Gaussians become more sharp. This result in further more lowering of the modeled probabilities.

C. Testing of HMM

HMM testing of speech commands starts with preprocessing and feature extraction. Then calculation for re-estimation of parameter was done with the help of Baum-Welch algorithm. Next step was to compare the probability score for each word model and choose the word with maximum score. Viterbi's algorithm is well appropriate for the recognition regime, when HMM is trained enough. Viterbi algorithm using dynamic programming is described in [3].Output of the above selected speech will be act as speech command, operating system will respond to speech command in the form of any operating system command activation.

IV. Experimental Results

The speech recognition for Hindi digits was developed for 10 digits (0 to 9) using HMM and the recognized digit was displayed in text format. The experimental program operates with standard WAV-files. By no special measures of optimizing the computing environment, the average training time for the whole database is about 5 minutes, i.e. about 3-4 sec per word version. Experiments with variation in method were conducted to evaluate the performance of our system. The first type of test was conducted using speakers involved in the training phase. The second type of test was conducted using speakers who were not involved in the training phase. The evaluation was made according to recognition accuracy of every word using only MFCC as feature vector and and the implemented method with time derivatives of MFCC.

A. Modeling framework

HMM Model is defined as : (Q,O, A,B,π)

where Q is $\{q_i\}$ (all possible states), O is $\{v_i\}$ (all possible observation) and X_t denotes the state at time t.

Q = 3 , O = 4

Transition matrix modeling:

A is $\{a_{ij}\}$ where $a_{ij} = P(X_{t+1} = q_j\,|X_t = q_i)$. The centre presents the mean of all cepstral vectors that the HMM "generates" being in that specific state. The covariance matrix is most often assumed diagonal one, i.e. that all cepstral vector components that are uncorrelated. Thus, each model is independently examined along its coordinates.

Observation probability matrix modeling:

B is $\{b_i\}$ where $b_i(k) = P(O_t = v_k|X_t = q_{it})$ (observation probabilities of observation k at state i).

1) For speech samples of speakers involved in the training phase:

Two different speakers who were part of the training phase were asked to utter at random, the ten different digits, thrice. The results are shown in Table I for each speaker.

Average Recognition accuracy: 73.33 %

Two different speakers who were not part of the training phase were asked to utter at random, the ten different digits, thrice. Recognition results for the second test in the range of 50-60%.

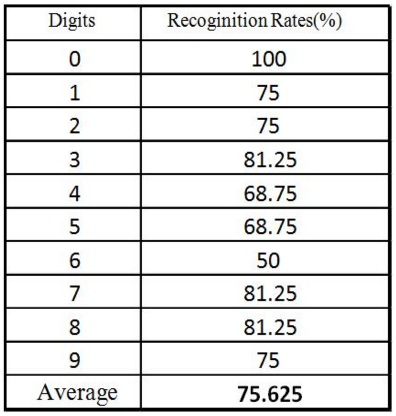
2) Table II gives the implemented method with time derivatives

of MFCC.

TABLE I. COMPARISON OF RECOGNITION RATES (%) [9]

Speakers	Nos. of spoken digits	Nos. of recognized digits	Recognition Rate
Speaker - 1	30	23	76.66
Speaker - 2	30	21	70

TABLE II. COMPARISON OF RECOGNITION RATES (%) FOR EACH DIGIT



Digits	Recoginition Rates(%)
0	100
1	75
2	75
3	81.25
4	68.75
5	68.75
6	50
7	81.25
8	81.25
9	75
Average	**75.625**

B. Analysis of results obtained

In this paper the experimental samples for training are recorded in laboratory (room) environment. In comparison to the results of research paper – [5] and [9], the results obtained are slightly better in normal setups. Besides the database size is deliberately taken to be small and supported with Vector quantization for training purpose. Since the re-estimation procedure is time consuming, the aim was to reduce processing time without compromising on the quality of recognition.

Note that the Recognition rate has decreased in the case due to the new speakers. Despite their widespread use, MFCCs are suboptimal. Their major flaw lies in their final calculation step, the inverse FFT; taking low-order cosine weightings of the log spectrum is motivated entirely on mathematical grounds unrelated to speech communication. The higher coefficients contain finer spectral detail, which altogether allow discrimination between similar sounds, but their lack of interpretation leave them highly vulnerable to non-ideal conditions such as noise or accents. Thus, when speech is corrupted by noise, which often fills the spectral valleys between harmonics and between formants, the MFCCs deteriorate. The above discussion on MFCC could be one of the reasons for lower recognition rate in the case of new speakers.

V. CONCLUSION AND FUTURE WORK

Analysis was done on database comprising of speech samples of different speakers. A method for adaptive and precise computation of probabilities modeled by Gaussian pdfs was implemented. The method aims an environment providing for convergence of the HMM training by Baum-Welch and the consecutive recognition by Viterbi algorithm. Final aim of this research was to develop a HMM based method for speech recognition with automatic setting up to the specifics of Hindi language [9]. The following can be concluded from detailed analysis of the results:

1) Words with more distinct phones produced less accuracy in recognition. Ex. The presence of plosives in the beginnings led to being misinterpreted as more than one word. As compared with the results of [9], the recognition rate of the words with more distinct phones shows marked improvement with the addition of temporal information to the extracted MFCC feature.

2) Words with similar type of phones were recognized interchangeably. Ex. The system got confused with words – 'saat, chaar and aat'.

3) The problem with emphasized plosives and unvoiced beginnings which was seen to be more prominent for Speaker 2, [9] was reduced by using DMFCC – DDMFCC as the feature.

4) 'Shunya' and 'teen' are very distinct from the rest of the digit pronunciations and show better recognition rates for both the speakers.

The above conclusion applies to both the feature extraction methods, i.e. MFCC and DMFCC – DDMFCC. The word wise recognition rate shows a overall improvement of approximately 2.5 % in the recognition rate of the system with the addition of temporal information to the extracted MFCC feature.

The most immediate scope of improvements are as listed below [9]:

1) Future work can be transition from word model to phoneme based model of the recognition system for further improvement.

2) In near future we intend to develop an approach for optimal choice of the HMM internal states number generally depending on the input word lengths and contents. A definite hope in this respect is for the almost

full representation of a set of about more number of Hindi allophones.

REFERENCES

[1] L. R. Rabiner. A tutorial on hidden markov models and selected applications in speech recognition. pages 256–286. Proceedings of the IEEE, 1989..

[2] Md. Rashidul Hasan, Mustafa Jamil, Md. Golam Rabbani Md. Saifur Rahman, Speaker Identification using Mel Frequency Cepstral Coefficients, 3rd International Conference on Electrical & Computer Engineering ICECE 2004, Dhaka, Bangladesh ISBN 984-32-1804-4 565.

[3] Pruthi T, Saksena, S and Das, P K (2000) Swaranjali: Isolated Word Recognition for Hindi Language using VQ and HMM. International Conference on Multimedia Processing and Systems (ICMPS), IIT Madras

[4] S.K. Podder, "Segment-based Stochastic Modeling for Speech Recognition". PhD Thesis. Department of Electrical and Electronic Engineering, Ehime University, Japan,1997.

[5] Ahmad A. M. Abushariah, Teddy S. Gunawan, Othman O. Khalifa, English Digits Speech Recognition System Based on Hidden Markov Models, International Conference on

Computer Communication Engineering (ICCCE 2010), Kuala Lumpur, Malaysia.

[6] Y. Linde, A. Buzo & R. Gray, "An algorithm for vector quantizer design", IEEE Transactions on Communications, Vol. 28, issue 1, Jan 1980 pp.84-95.

[7] Kuldeep Kumar,R. K. Aggarwal, Hindi speech recognition system using HTK, International Journal of Computing and Business Research ISSN (Online) : 2229-6166 Volume 2 Issue 2 May 2011.

[8] Ch.Srinivasa Kumar, Dr. P. Mallikarjuna Rao, Recognition System Using MFCC, Vector Quantization And LBG Algorithm, International Journal on Computer Science and Engineering (IJCSE), Vol. 3 No. 8 August 2011.

[9] Sanghamitra V. Arora, Prof.(Mrs) R.A. Deshpande, Modeling and recognition of Isolated words using VQ and Hidden Markov Model, Second International Conference on Computational Intelligence and Information Technology – CIIT 2012, ACEEE, Chennai, India.

[10] A Novel Approach for MFCC Feature Extraction, Md. Afzal Hossan, Sheeraz Memon, Mark A Gregory, 4 th International Conference on Signal Processing and Communication Systems (ICSPCS), December 2010, Melbourne, E-ISBN : 978-1-4244-7906-1.

[11] Delta-spectral cepstral coefficients for robust speech recognition, Kshitiz Kumar, Chanwoo Kim and Richard M. Stern, International Conference on Acoustics, Speech and Signal Processing (ICASSP), May 2011, Prague, ISSN : 1520-6149.

DOI: 01.IJSIP.4.3.1257